CryptoExperts
WE INNOVATE TO SECURE YOUR BUSINESS

RÉPUBLIQUE FRANÇAISE
Liberté
Égalité
Fraternité

AGENCE NATIONALE DE LA SÉCURITÉ DES SYSTÈMES D'INFORMATION
ANSSI

CRY.ME : un challenge de cryptographie sur une messagerie sécurisée

Sonia Belaïd[2], Ryad Benadjila[1, 2], Thibauld Feneuil[2], Jérémy Jean[1], Louiza Khati[1], Ange Martinelli[1], **Chrysanthi Mavromati**[1], **Abdul Rahman Taleb**[2] and Matthieu Rivain[2]

[1] ANSSI, [2] CryptoExperts

SSTIC - 7 juin 2023

Designing CRY.ME
(CRYptographic MEssaging application)

Why CRY.ME?

ANSSI - French certification center

- Implements the certification scheme
- Licenses the ITSEFs (Information Technology Security Evaluation Facilities) which conduct the evaluations
  - Audit (preliminary and follow-up)
  - Follow-up of their technical skills through evaluations
  - Mandatory participation to technical challenges
- Supervises the evaluation projects

2022: CRY.ME challenge for all ITSEFs and only for cryptographic analysis

Challenge Design

- Cryptographic analysis compliant to ANSSI's procedures[1, 2]
- Test vehicle common to all ITSEFs (software and hardware) but not only
- Cover vulnerabilities of different types and levels of difficulty

[1] ANSSI-CC-CRY-P-01 : https://www.ssi.gouv.fr/uploads/2014/11/anssi-cc-cry-p-01-modalites-pour-la-realisation-des-analyses-cryptographiques_v4.1.pdf

[2] ANSSI-PG-083 : https://www.ssi.gouv.fr/uploads/2021/03/anssi-guide-mecanismes_crypto-2.04.pdf

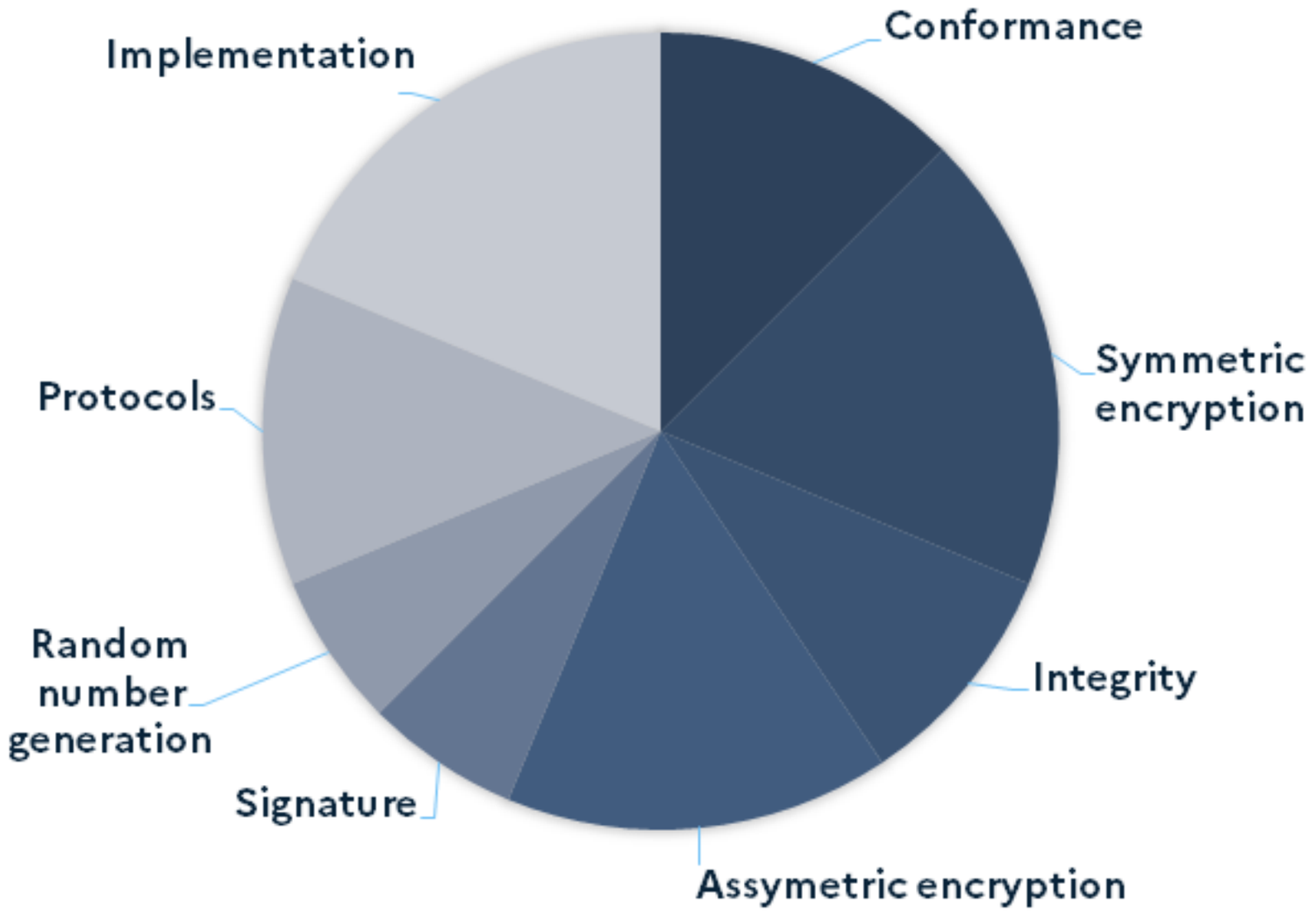
Challenge Design

- Cryptographic analysis compliant to ANSSI's procedures[1, 2]
- Test vehicle common to all ITSEFs (software and hardware) but not only
  - **Open-source software product**
- Cover vulnerabilities of different types and levels of difficulty
  - **Several types of vulnerabilities:** conformance, symmetric crypto, asymmetric crypto, random number generation, protocols, implementation
  - **Three levels of difficulty:** not necessarily the same difficulty for identification and exploitation
  - **Various sources of errors:** weak choice of algorithms, design and/or implementation errors, differences between specifications and implementation, etc.

[1] ANSSI-CC-CRY-P-01 : https://www.ssi.gouv.fr/uploads/2014/11/anssi-cc-cry-p-01-modalites-pour-la-realisation-des-analyses-cryptographiques_v4.1.pdf

[2] ANSSI-PG-083 : https://www.ssi.gouv.fr/uploads/2021/03/anssi-guide-mecanismes_crypto-2.04.pdf

Vulnerabilities



Example 1: AES

- AES implementation that does not respect the standard

```
243   // --- AES BLOCK FUNCTIONS
244   // ----------------------------------------------------------------------------
245
246   void aes256_encrypt(const BYTE in[], BYTE out[], const WORD exp_key[])
247   {
248       BYTE state[AES_BLOCK_SIZE];
249       int r;
250
251       // init state
252       memcpy(state, in, AES_BLOCK_SIZE);
253
254       add_round_key(state, exp_key);
255
256       for(r=0; r<NB_ROUNDS; r++)
257       {
258           sub_bytes(state);
259           shift_rows(state);
260           mix_columns(state);
261           add_round_key(state, exp_key+r*4);
262       }
263
264       sub_bytes(state);
265       shift_rows(state);
266       add_round_key(state, exp_key+NB_ROUNDS*4);
267
268       // output final state
269       memcpy(out, state, AES_BLOCK_SIZE);
270   }
```

Example 2: RSA-512 Signature

- Impersonates server's identity
- Factorisation takes ~10 days using a common laptop

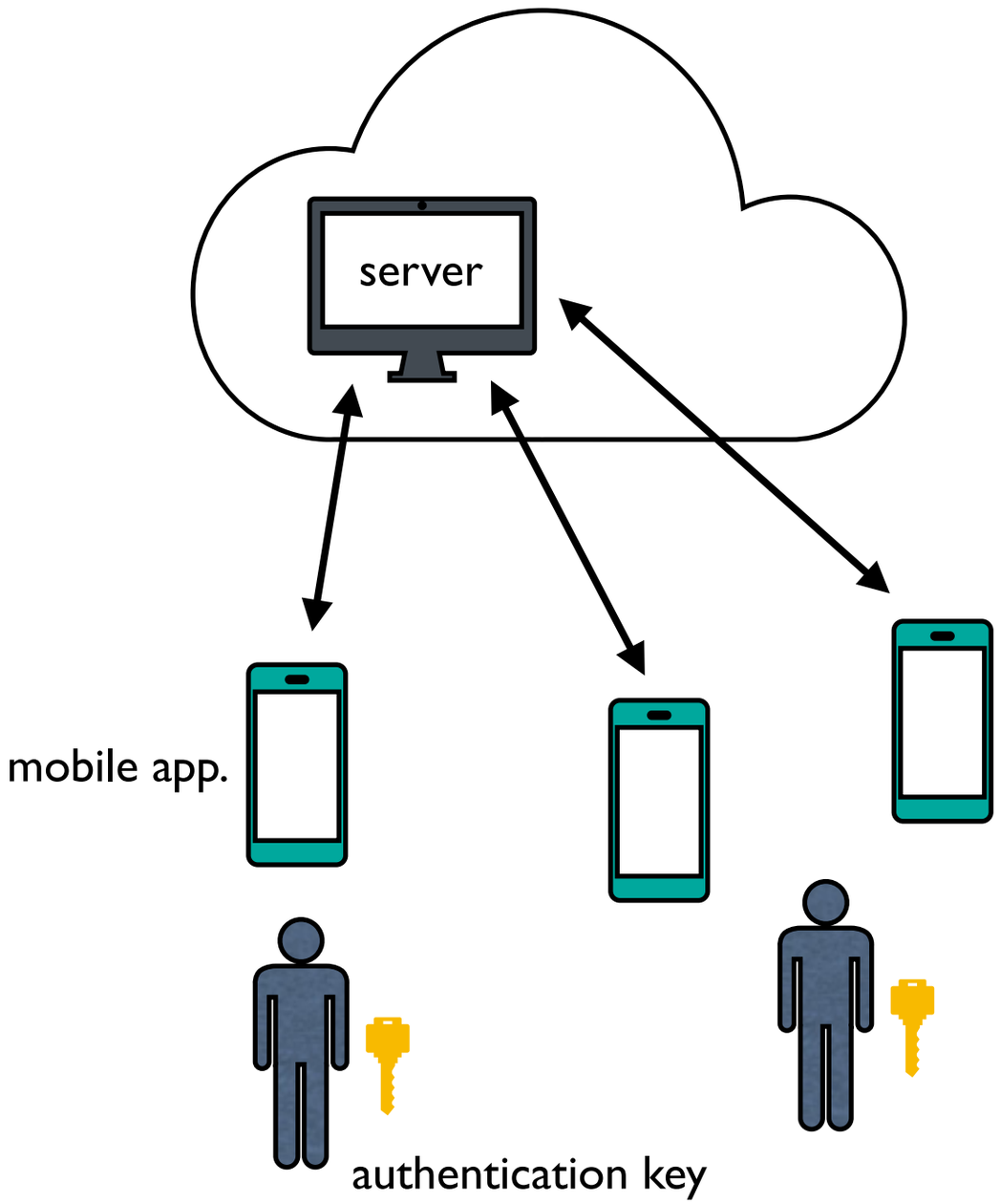
```
17    #define RSA_ACCREDITATION_MOD_LENGTH 64
18    #define RSA_ACCREDITATION_HASH_LENGTH 20
```

CRY.ME « Element » Application

Why Element?

- Open-source Android application
- Integrates Matrix end-to-end encryption
- Integrates secure data backup
- Supports group conversations
- Convenient code modification and navigation (Yubikey integration, …)
- Easy-to-use Interface
- No external dependencies besides matrix android SDK

Architecture



Functionalities

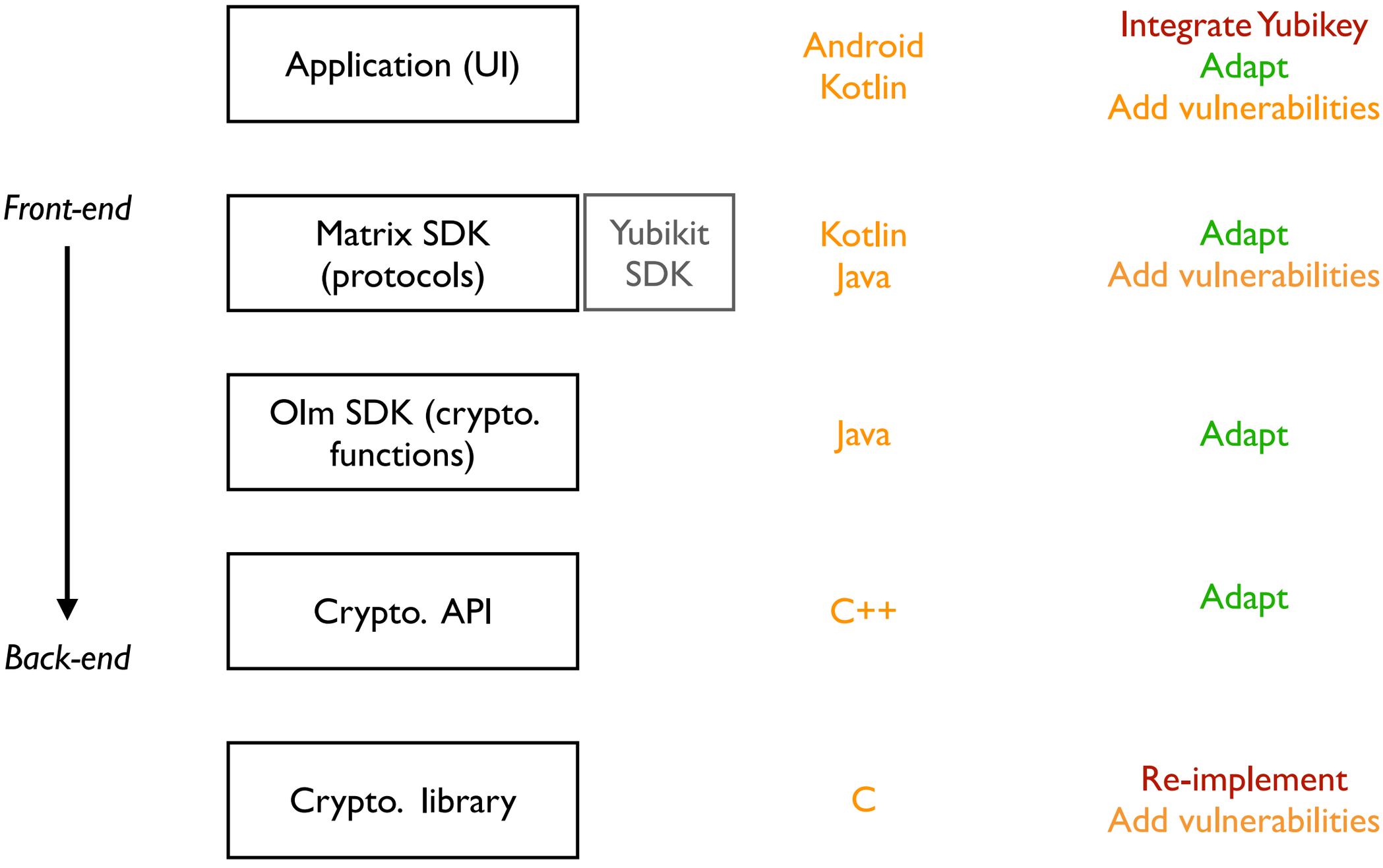
- Create account
- Login / Logout
- Send / Receive messages (1-to-1 or group)
- Send / Receive attachments
- Secure backup storage on the server
- Out-of-band verification between users

What we offer

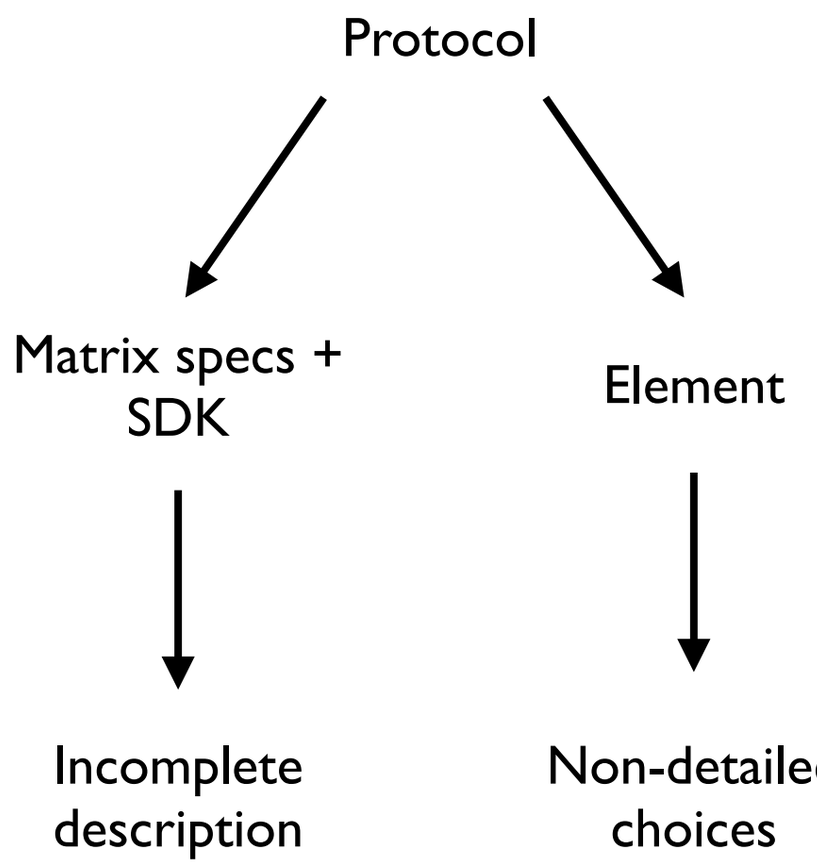
- Open-source code
- **Modified** Element code with/without vulnerability comments
- Server code running in a Docker container
- Emulator code running on a desktop and handling authentication keys and server connections
- Documents (in French) describing security specifications and targets

Challenge Setup

Setup of Different Layers



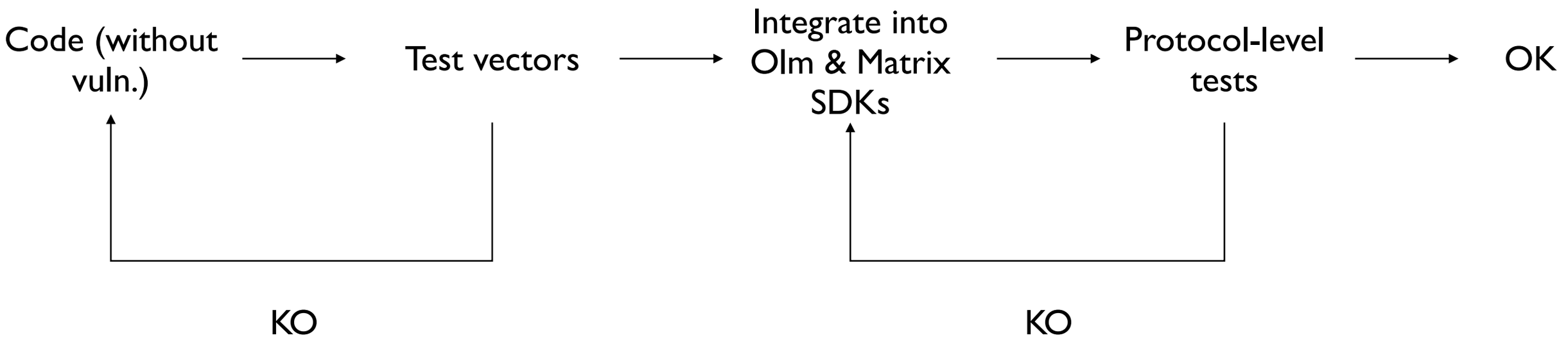
Cryptographic Protocols



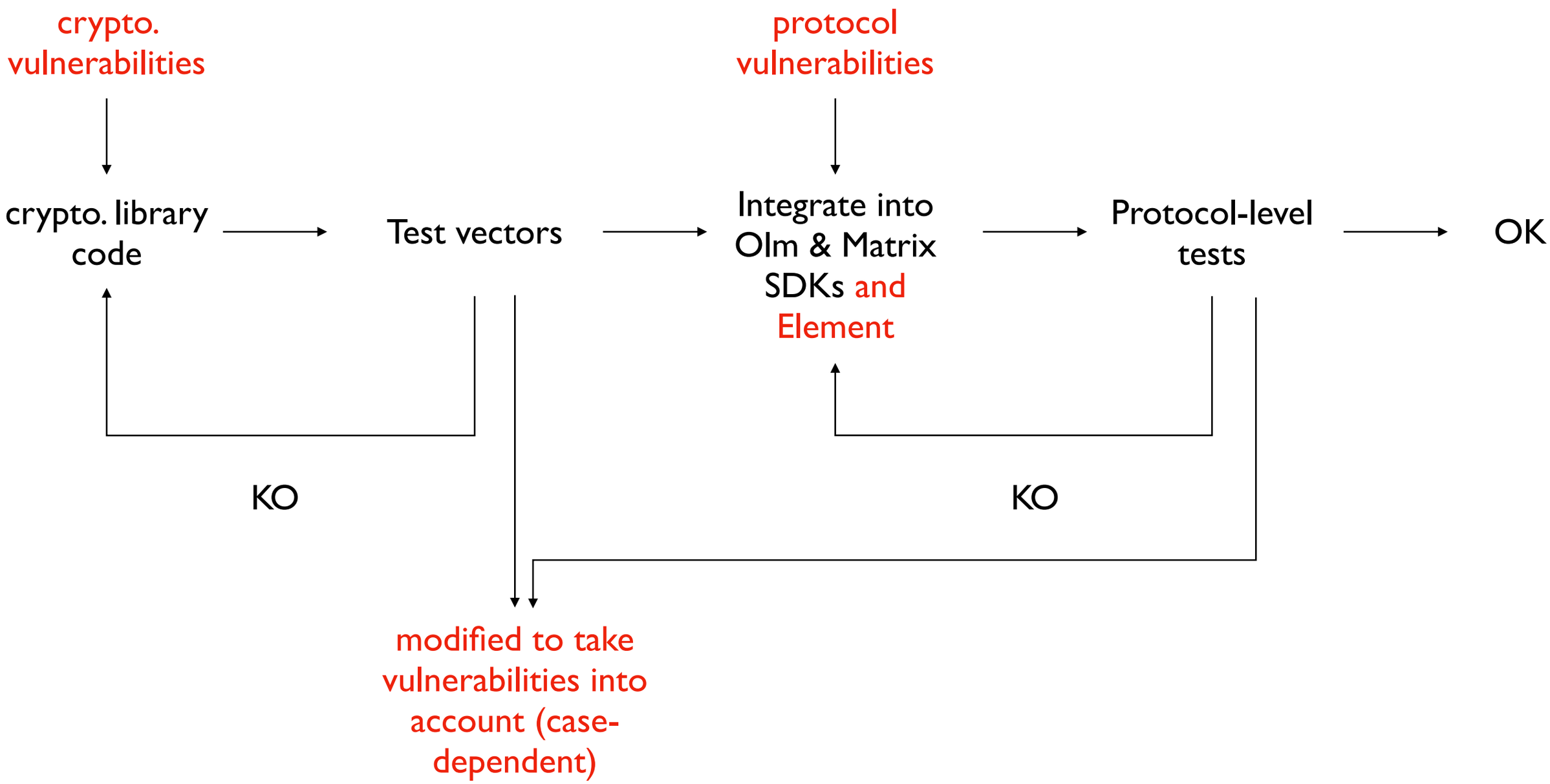
- Goal: provide a complete comprehensive description from both sources for each protocol

Implementation of Cryptographic Primitives

- Matrix uses Olm SDK, implementing all crypto. algorithms
- CRY.ME: re-implementing all crypto. algorithms from scratch
- Process must ensure application's functionality



Integration of Vulnerabilities



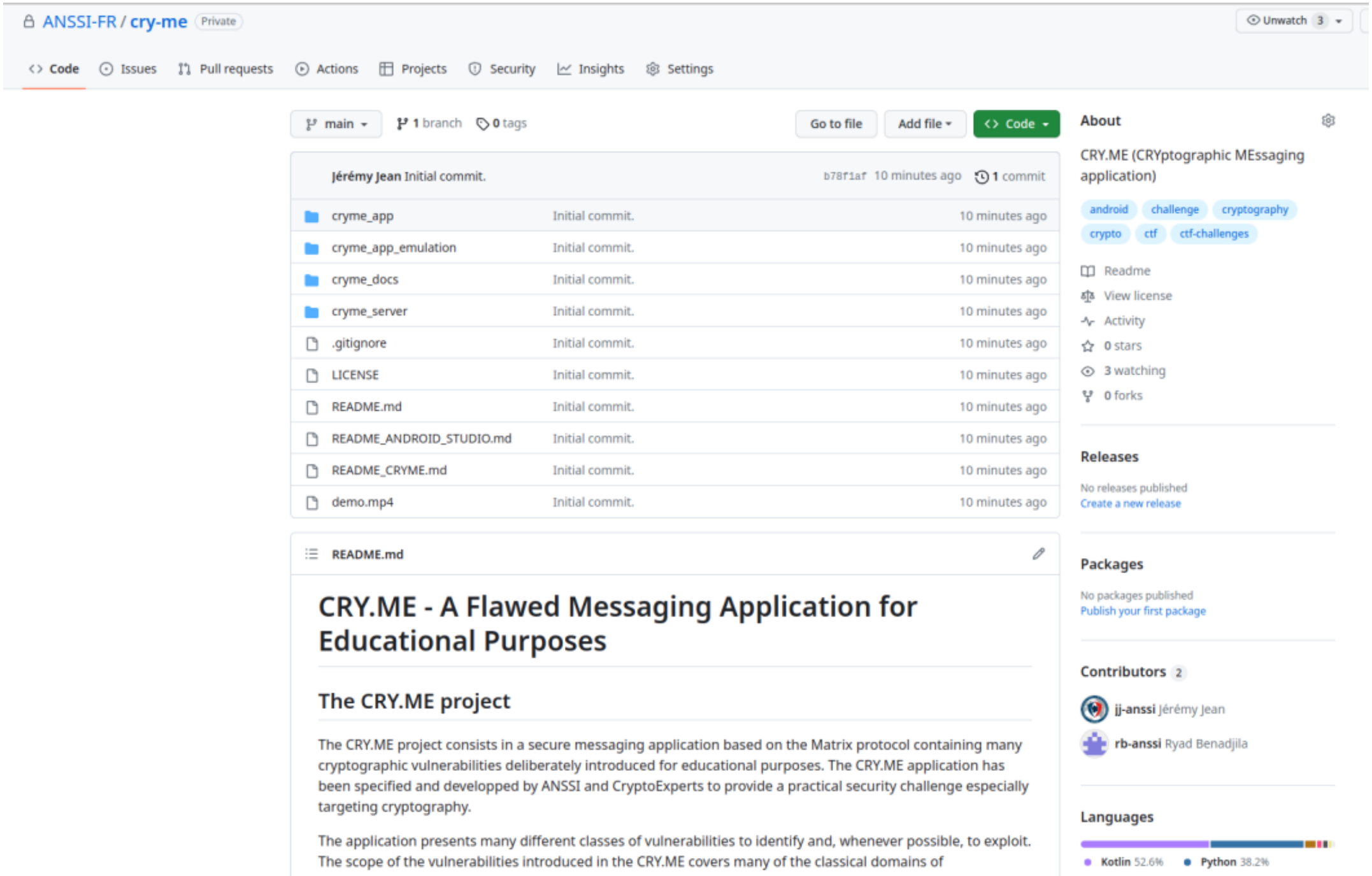
Server Setup

- 3-layer architecture
- Nginx
  - handles TLS
- Synapse
  - handles app. functionalities
- PostgreSQL
  - handles database

Conclusion

- Many **cryptographic vulnerabilities** to find
- **Open-source** material with a guided tutorial
  - https://github.com/ANSSI-FR/cry-me
- The challenge serves as a tool for **learning purposes** on
  - the impact of cryptographic vulnerabilities on a known messaging application
  - how to detect them
  - if and how to exploit them

Questions ?



ANSSI-FR / cry-me Private

Unwatch 3

Code Issues Pull requests Actions Projects Security Insights Settings

main 1 branch 0 tags Go to file Add file Code

Jérémy Jean Initial commit. b78f1af 10 minutes ago 1 commit

cryme_app	Initial commit.	10 minutes ago
cryme_app_emulation	Initial commit.	10 minutes ago
cryme_docs	Initial commit.	10 minutes ago
cryme_server	Initial commit.	10 minutes ago
.gitignore	Initial commit.	10 minutes ago
LICENSE	Initial commit.	10 minutes ago
README.md	Initial commit.	10 minutes ago
README_ANDROID_STUDIO.md	Initial commit.	10 minutes ago
README_CRYME.md	Initial commit.	10 minutes ago
demo.mp4	Initial commit.	10 minutes ago

README.md

CRY.ME - A Flawed Messaging Application for Educational Purposes

The CRY.ME project

The CRY.ME project consists in a secure messaging application based on the Matrix protocol containing many cryptographic vulnerabilities deliberately introduced for educational purposes. The CRY.ME application has been specified and developped by ANSSI and CryptoExperts to provide a practical security challenge especially targeting cryptography.

The application presents many different classes of vulnerabilities to identify and, whenever possible, to exploit. The scope of the vulnerabilities introduced in the CRY.ME covers many of the classical domains of

About

CRY.ME (CRYptographic MEssaging application)

android challenge cryptography crypto ctf ctf-challenges

Readme
View license
Activity
0 stars
3 watching
0 forks

Releases

No releases published
Create a new release

Packages

No packages published
Publish your first package

Contributors 2

jj-anssi Jérémy Jean
rb-anssi Ryad Benadjila

Languages

Kotlin 52.6% Python 38.2%